

A Bipartite Graph Matching Framework for Finding Correspondences between Structural Elements in Two Proteins*

Yuhang Wang, Fillia Makedon, James Ford

Department of Computer Science, Dartmouth College, Hanover, NH 03755, USA

Abstract—A protein molecule consists one or more chains of amino acid sequences that fold into a complex three-dimensional structure. A protein’s functions are often determined by its 3D structure, and so comparing the similarity of 3D structures between proteins is an important problem. To accomplish such comparison, one must align two proteins properly with rotation and translation in 3D space. Finding the correspondences between structural elements in the two proteins is the key step in many protein structure alignment algorithms. In this paper, we introduce a new graph theoretic framework based on bipartite graph matching for finding sufficiently good correspondences. It is capable of providing both sequence-dependent and sequence-independent correspondences. It is a general framework for pair-wise matching of atoms, amino acids residues or secondary structure elements.

Keywords— Protein structure alignment, bipartite graph matching, correspondence

I. INTRODUCTION

Proteins are the macromolecules that carry out most of the essential functions in living cells. A protein is composed of long trains of amino acids linked together; the amino acid sequence of a protein is called its primary structure. There are twenty amino acids that are commonly found in proteins, each with a similar, yet unique structure. Certain local portions of an amino acid sequence fold into particular shapes called secondary structure elements (SSE), the most common of which are the alpha helix and the beta strand. Protein properties are determined by the 3D configuration, or tertiary structure, of a protein. Proteins achieve their biological functions by binding to other molecules, and the tertiary structure controls the existence and placement of binding sites. Therefore, the 3D structures of protein are extremely important. Proteins with similar 3D structures typically have similar functions. In evolutionary related proteins, 3D structure is much better preserved than sequence.

The Protein Data Bank (PDB) [1, 2] is a centralized database that contains protein structures determined experimentally, either by X-ray crystallography or NMR spectroscopy. Information about each protein is stored in a PDB file and mmCIF (macromolecular Crystallographic Information File) file [3]. Stored information includes the primary sequence, 3D coordinates of all the atoms,

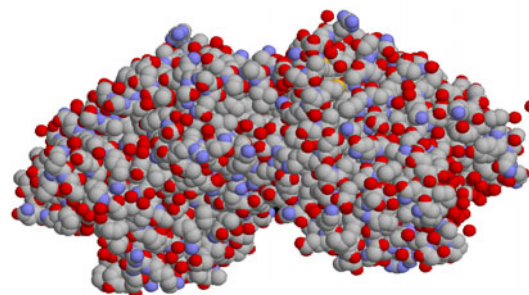


Fig. 1. 3D rendition of the protein with PDB ID “1cpc”

information about SSEs, and other metadata. The latter format is the most recent standard. Figure 1 shows the visualization of the 3D structure of the protein “1cpc” in the Protein Data Bank.

To compare the 3D structures of two proteins, one must align them properly in 3D space. Protein structure alignment is a very important subject [4] because it enables us to learn about the functional relationships between proteins, protein homology and structural motifs (common building blocks in a group of proteins), and also because it helps in structure prediction. Furthermore, data mining tasks such as protein clustering and classification also rely on structure alignment to provide similarity measures.

Structure alignment amounts to optimally aligning two structures through rotation and translation so that a certain objective function is minimized. The commonly used objective function is the Root Mean Square Distance (RMSD) [5] between corresponding atom pairs. Whereas sequence alignment can be solved within polynomial time by using dynamic programming methods [6], this is not the case for structure alignment. All protein structure alignment algorithms give locally optimal, and thus approximate, solutions.

The key problem in protein structure alignment is to find the optimal correspondence between the atoms in the two proteins. The goal is to determine which atoms in one protein correspond to those in the other. More formally, given two protein structures with elements:

$$A = (a_1, a_2, \dots, a_n),$$

$$B = (b_1, b_2, \dots, b_m),$$

A *correspondence* is a set of pairs

$$C(A, B) = \{(a_{i_1}, b_{j_1}), (a_{i_2}, b_{j_2}), \dots, (a_{i_k}, b_{j_k})\}.$$

The optimal correspondence should lead to the smallest RMSD between the two structures. An exhaustive search is

*This work was supported in part by the National Science Foundation under grant ITR-0312629.

computationally intractable. However, given such a correspondence, the problem of optimally aligning two structures through rotation and translation so that the RMSD is minimized can be solved efficiently in time linear in the number of atoms [7].

In this paper, we introduce a new graph theoretic framework based on bipartite graph matching [8] for finding sufficiently good correspondences. It is capable of providing both sequence-dependent and sequence-independent correspondences. It is a general framework for pair-wise matching of atoms, amino acids residues or SSEs.

II. RELATED WORK

Many approaches have been proposed for protein structure alignment, such as DALI [9], STRUCTAL [10], VAST [11, 12], LOCK [13], MUSTA [14, 15], CE [16], *etc.* One important framework is based on the iterative dynamic programming method originated by Rossmann *et al.* [17, 18], where one first computes a distance matrix between all pairs of atoms to form a similarity matrix, which by dynamical programming methods gives rise to an optimal correspondence mimicking the sequence alignment procedure [6]. It works as follows:

To align A and B

1. $C =$ Initial correspondence
2. Compute the optimal rotation and translation of A to minimize the RMSD between A and B, and superpose A and B accordingly
3. Find the new correspondence using dynamic programming based on the new superposition
4. If $C' \neq C$, then $C = C'$, go to 2.

Many different approaches have been proposed under this framework [10, 13, 16, 19, 20], either using it directly or as a refinement step. The output of this framework depends significantly on the quality of the initial correspondence [21]. Therefore, a reliable way of finding a good correspondence is needed. In fact, the iterative dynamic programming framework is very similar to the Iterative Closest Points algorithm [22] in computer vision.

Most methods, except those based on geometry hashing [14, 15], are sequence dependent. However, amino acids far apart in the primary sequence may be brought together in 3D space to form the binding sites of proteins. Molecular evolution studies [23] also show that as a protein molecule evolves, its structure inclines to be similar to its ancestor, even when the sequence might have changed significantly. Sequence-independent alignment enables detection of non-sequential motifs in proteins, especially similar binding sites. However, neglecting sequential order information also has disadvantages. For example, motifs preserving sequence order might be biologically more meaningful than nonsequential motifs of similar size. Thus it is advantageous

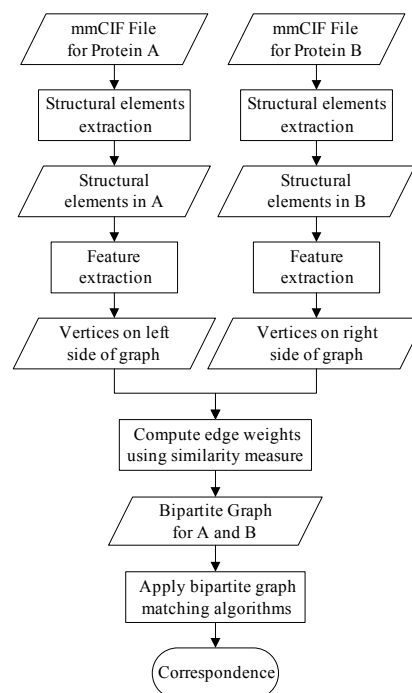


Fig. 2. Overview of the proposed framework

to be able to compute both sequence-dependent and sequence-independent correspondences. Our proposed framework based on bipartite graph matching can compute both types of correspondences.

Bipartite matching methods have also been applied in image feature matching [24]. Other graph theoretic approaches for protein structure alignment also exist [12, 25], which are based on max clique detection. Since the max clique problem is known to be NP-hard, it is only feasible for small graphs with about less than 30 vertices.

III. A NEW BIPARTITE MATCHING-BASED FRAMEWORK

A. System Overview

Figure 2 illustrates the proposed framework based on bipartite graph matching. First we extract structural elements from each protein data file. Feature vectors are then computed for the structural elements, forming the vertices for a bipartite graph. The vertices on the left and right side of the graph represent the structural elements in protein A and protein B respectively. Each vertex is connected to all the vertices on the opposite side. The weight on an edge is the output of a specially designed similarity measure taking as inputs the two feature vectors for the structural elements represented by the two vertices incident to the edge. We then apply bipartite graph matching algorithms on this graph to obtain the set of edges in the optimal matching, which represents the correspondence.

B. Reduction to a weighted bipartite graph

The extracted structural elements can be all atoms, the central carbon atoms (C_α atoms) of the amino acids, amino acids residues, or secondary structures. The feature vector for each structural element can be derived from geometric properties (e.g., position of atoms and side-chain orientation) and chemical properties (e.g., residue type and charges).

Given the two sets of structure elements $A = (a_1, a_2, \dots, a_n)$ from protein A and $B = (b_1, b_2, \dots, b_m)$ from protein B, an undirected weighted bipartite graph $G = (V, E)$ can be constructed as follows: $V = A \cup B, E = \{e_{ij}\}$. Each edge e_{ij} ($i = 1, 2, \dots, n; j = 1, 2, \dots, m$) corresponds to a weighted link between a_i and b_j , whose weight $w(e_{ij})$ is equal to the similarity between a_i and b_j , i.e. $w(e_{ij}) = \text{sim}(\vec{f}_{a_i}, \vec{f}_{b_j})$, where \vec{f}_{a_i} and \vec{f}_{b_j} are the feature vectors for a_i and b_j , respectively. A graph is bipartite if it has two kinds of vertices and the edges are only allowed between vertices of different kind. Obviously, the graph thus created is a weighted bipartite graph by construction.

C. Bipartite Graph Matching

A *matching* in a graph $G = (V, E)$ is a subset M of the edges E such that no two edges in M share a common vertex. We use $G = (V = A \cup B, E)$ to denote a bipartite graph, where A and B are the two kinds of vertices and E denotes the edges of G . A matching in a bipartite graph assigns vertices of A to vertices of B . A *maximum cardinality matching* is a matching with the maximum number of edges. If the edges of the graph have associated weights, then a *maximum weight matching* is a matching such that the sum of the weights of the edges in the matching is maximized. A *maximum weight maximum cardinality matching* is a maximum cardinality matching with the greatest weight. Figure 3 shows an example of the maximum weight bipartite matching.

In the context of protein structural elements correspondence, a maximum weight matching would return the correspondence with the maximum weight, but there is no guarantee of maximum cardinality. Therefore, some elements in the smaller protein may not be matched to any element in the other protein. In other words, a maximum weight matching favors good local matches.

A maximum weight maximum cardinality matching, on the other hand, would always return the matching with maximum cardinality, even if some edges in the matching

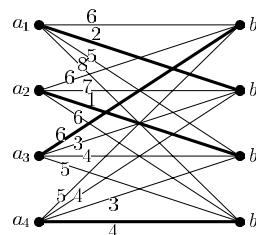


Fig. 3. Example of maximum weight bipartite matching. The thick edges denote the maximum weight matching.

have relatively small weights. It guarantees that every element in the smaller protein will be matched to an element in the other protein. In other words, a maximum weight maximum cardinality matching favors good global matches.

The best known strongly polynomial time bound algorithm for weighted bipartite matching is the classical Hungarian method due to Kuhn [26], which runs in time $O(|V|(|E| + |V| \log |V|))$. Weighted bipartite matching algorithms can be implemented efficiently, and can be applied to graphs of reasonably large size (about 100,000 vertices) [27].

D. Sequence Order Preservation

Traditional bipartite graphing matching algorithms do not preserve the sequence order of protein structural elements. As discussed in Section II, both sequence-dependent and sequence-independent correspondences have advantages and disadvantages. If sequence order dependence is desired, we can solve the matching problem with dynamic programming based on the following recurrence formula.

$$M_{ij} = w(e_{ij}) + \max \left(\begin{array}{l} w(e_{i-1,j-1}), \\ \max(M_{i',j-1}, i' < i \text{ and } e_{i',j-1} \in E), \\ \max(M_{i-1,j'}, j' < j \text{ and } e_{i-1,j'} \in E) \end{array} \right)$$

where

M_{ij} is the total weight for the matching ending at a_i and b_j .

It is easy to see that this recurrence can be solved in time $O(|A| \cdot |B| \cdot |E|)$ with straightforward application of the dynamic programming technique. To obtain the actual correspondence, one need only to keep track of the matched elements with pointers and follow the pointers from the last matched pair.

IV. DISCUSSION AND ONGOING WORK

In this paper, we have proposed a new framework for finding the correspondences between structural elements in

two proteins. Our framework is based on bipartite graph matching. Unlike the max-clique based methods previously used in protein structure alignment, bipartite graph matching can be computed efficiently in polynomial time, and the output matching is globally optimal. Our framework can be used in two ways:

- as a full protein structure alignment method: by feeding the correspondence computed by this framework to the algorithm in [7], we can align the two structures through rotation and translation so that the RMSD is minimized.
- for finding the initial seed alignment for other methods (methods that use iteration or clustering).

We have implemented our framework in C++ and Java using MBT [28] and LEDA [29]. In the current implementation, we used C_α atoms of the amino acids as structural elements. The feature vector for each C_α atom is the 3D R-histogram for that atom with respect to all other atoms in the protein structure. The 3D R-histogram is extended from the 2D version [30]. We are currently evaluating the performance of difference similarity measures with all pairs of proteins in the PDB. We will also examine the performance of alternative choices of structural elements, such as SSEs.

ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation under grant ITR-0312629. The authors thank Dr. Lincong Wang, Dr. Robert L. Drysdale, and Dr. Raul Covian for their valuable discussions.

REFERENCES

- [1] F. C. Bernstein, T. F. Koetzle, G. J. Williams, E. F. Meyer, Jr., M. D. Brice, J. R. Rodgers, O. Kennard, T. Shimanouchi, and M. Tasumi, "The Protein Data Bank: a computer-based archival file for macromolecular structures," *J Mol Biol*, vol. 112, pp. 535-42, 1977.
- [2] "Protein Data Bank, <http://www.rcsb.org/pdb/>."
- [3] J. D. Westbrook and P. M. Fitzgerald, "The PDB format, mmCIF, and other data formats," *Methods Biochem Anal*, vol. 44, pp. 161-179, 2003.
- [4] I. Eidhammer, I. Jonassen, and W. R. Taylor, "Structure comparison and structure patterns," *J Comput Biol*, vol. 7, pp. 685-716, 2000.
- [5] P. Koehl, "Protein structure similarities," *Current Opinion in Structural Biology*, vol. 11, pp. 348-353, 2001.
- [6] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *J Mol Biol*, vol. 48, pp. 443-53, 1970.
- [7] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, pp. 698-700, 1987.
- [8] L. Lovasz and M. D. Plummer, *Matching theory*. Amsterdam ; New York, N.Y.: North-Holland, 1986.
- [9] L. Holm and C. Sander, "Protein Structure Comparison by Alignment of Distance Matrices," *Journal of Molecular Biology*, vol. 233, pp. 123-138, 1993.
- [10] M. Gerstein and M. Levitt, "Comprehensive assessment of automatic structural alignment against a manual standard, the scop classification of proteins," *Protein Sci*, vol. 7, pp. 445-56, 1998.
- [11] J.-F. Gibrat, T. Madej, and S. H. Bryant, "Surprising similarities in structure comparison," *Current Opinion in Structural Biology*, vol. 6, pp. 377-385, 1996.
- [12] T. Madej, J. F. Gibrat, and S. H. Bryant, "Threading a database of protein cores," *Proteins*, vol. 23, pp. 356-69, 1995.
- [13] A. P. Singh and D. L. Brutlag, "Hierarchical protein structure superposition using both secondary structure and atomic representations," presented at Proc Int Conf Intell Syst Mol Biol, 1997.
- [14] N. Leibowitz, Z. Y. Fligelman, R. Nussinov, and H. J. Wolfson, "Automated multiple structure alignment and detection of a common substructural motif," *Proteins*, vol. 43, pp. 235-45, 2001.
- [15] R. Nussinov and H. J. Wolfson, "Efficient detection of three-dimensional structural motifs in biological macromolecules by computer vision techniques," *Proc Natl Acad Sci U S A*, vol. 88, pp. 10495-10499, 1991.
- [16] J. Rose and F. Eisenmenger, "A fast unbiased comparison of protein structures by means of the Needleman-Wunsch algorithm," *J Mol Evol*, vol. 32, pp. 340-54, 1991.
- [17] S. T. Rao and M. G. Rossmann, "Comparison of super-secondary structures in proteins," *J Mol Biol*, vol. 76, pp. 241-56, 1973.
- [18] M. G. Rossmann and P. Argos, "Exploring structural homology of proteins," *J Mol Biol*, vol. 105, pp. 75-95, 1976.
- [19] L. Holm and C. Sander, "3-D lookup: fast protein structure database searches at 90% reliability," *Proc Int Conf Intell Syst Mol Biol*, vol. 3, pp. 179-87, 1995.
- [20] R. Blankenbecler, M. Ohlsson, C. Peterson, and M. Ringner, "Matching protein structures with fuzzy alignments," *Proc Natl Acad Sci U S A*, vol. 100, pp. 11936-40, 2003.
- [21] Z. K. Feng and M. J. Sippl, "Optimum superimposition of protein structures: ambiguities and implications," *Fold Des*, vol. 1, pp. 123-32, 1996.
- [22] P. J. Besl and H. D. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 239-256, 1992.
- [23] W.-H. Li, *Molecular evolution*. Sunderland, Mass.: Sinauer Associates, 1997.
- [24] Y. Cheng, V. Wu, R. Collins, A. Hanson, and E. Riseman, "Maximum-Weight Bipartite Matching Technique and Its Application in Image Feature Matching," presented at SPIE Conference on Visual Communication and Image Processing, 1996.
- [25] P. J. Artymiuk, A. R. Poirrette, H. M. Grindley, D. W. Rice, and P. Willett, "A graph-theoretic approach to the identification of three-dimensional patterns of amino acid side-chains in protein structures," *J Mol Biol*, vol. 243, pp. 327-44, 1994.
- [26] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, pp. 83-97, 1955.
- [27] D. S. Johnson and C. McGeoch, "Network Flows and Matching: First DIMACS Implementation Challenge," American Mathematical Society, 1993.
- [28] "The Molecular Biology Toolkit (MBT)," 1.0.0 ed. San Diego Supercomputer Center: <http://mbt.sdsc.edu/>, 2004.
- [29] K. Mehlhorn and S. Näher, *LEDA: a platform for combinatorial and geometric computing*. Cambridge, U.K. ; New York: Cambridge University Press, 1999.
- [30] Y. Wang and F. Makedon, "R-Histogram: Quantitative Representation of Spatial Relations for Similarity-Based Image Retrieval," presented at The 11th Annual ACM International Conference on Multimedia, Berkeley, California, USA, 2003.